

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

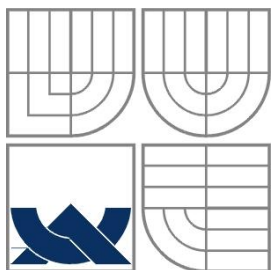
## REKONSTRUKCE POZADÍ Z NĚKOLIKA FOTOGRAFIÍ

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

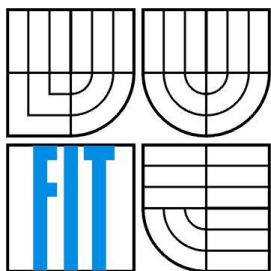
AUTOR PRÁCE  
AUTHOR

MARTIN HREBÍČEK

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

# REKONSTRUKCE POZADÍ Z NĚKOLIKA FOTOGRAFIÍ

BACKGROUND RECONSTRUCTION FROM SEVERAL PHOTOGRAPHS

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

MARTIN HREBÍČEK

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. MICHAL SEEMAN

BRNO 2010

## **Abstrakt**

Tato práce se zabývá rekonstrukcí pozadí z několika fotografií. Klade si za cíl odstranit z fotografií nežádoucí objekty, které mohou například představovat lidi, zvířata anebo dopravní prostředky. Porovnává a opisuje různé algoritmy jak problematiku řešit. Práce dále obsahuje implementaci vybraných algoritmů a jejich testování.

## **Abstract**

This thesis deals with background reconstruction from several photographs. It is focused on removing undesirable objects, which for example could be humans, animals or means of transport. It compares and describes different algorithms how to solve problems. The thesis also consists of the implementation of chosen algorithms and its testing.

## **Klíčová slova**

Fotografie, rekonstrukce pozadí, osvětlení scény

## **Keywords**

Photography, background reconstruction, scene light effects

## **Citace**

Hřebíček Martin: Rekonstrukce pozadí z několika fotografií, bakalářská práce, Brno, FIT VUT v Brně, 2010

# Rekonstrukce pozadí z několika fotografií

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Michala Seemana.

.....  
Martin Hřebíček  
26. 4. 2010

## Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Ing. Michalu Seemanovi za pomoc a cenné rady.

© Martin Hřebíček, 2010

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

Obsah.....	1
1 Úvod.....	2
2 Fotografia .....	3
2.1 Pozadie .....	3
2.2 Vyhľadavanie určitých objektov v obraze.....	3
2.3 Použitie.....	4
3 Problémy pri rekonštrukcii pozadia .....	5
3.1 Počasie.....	5
4 Vybrané algoritmy .....	7
4.1 An improved adaptive background mixture model for realtime tracking with shadow detection .....	8
4.2 Object removal by exemplar based inpainting.....	9
5 Implementácia .....	10
5.1 Pamäťové štruktúry.....	10
5.2 Funkcie rekonštrukcie pozadia .....	11
5.2.1 Priemer .....	11
5.2.2 Rozmedzie .....	11
5.2.3 Kocka.....	12
5.2.4 Susedia.....	14
5.3 Výsledný postup .....	16
6 Testovanie .....	18
6.1 Rozmedzie .....	20
6.2 Kocka .....	21
6.3 Susedia .....	23
7 Záver .....	25

# 1 Úvod

Fotografia slúži ako vizuálna pamiatka určitého momentu. Za predchodcu možno považovať maľbu. Spomedzi rôznych druhov sa sústredíme na zachytenie statického obrazu. Každý videl umelecké dielo, na ktorom je príroda alebo ulica, prípadne budova samotná. Panovníci si nechávali zvečniť okrem svojej rodiny aj svoje sídla vo forme obrazov. Umelec pri tvorbe diela hľadel na objekt a maľoval čo videl. Prípadné nežiaduce vplyvy si sám odfiltroval. Ak sa prechádzali ľudia pred palácom, tak ich mohol namaľovať na ľubovoľnom mieste. Obraz by dotvorili a vylepšili. Mohol ich aj vynechať, ak uznal za vhodné.

V súčasnosti je najjednoduchšie si predmet záujmu odfotiť. Často sa však na výslednej fotografii nachádza aj objekt, ktorý je na nej nežiadaný. Turista stojí pred pamiatkou v centre mesta. S daným záujmom nie je sám. Každý si chce zachovať spomienku o tom, že pamätihodnosť navštívil. Či už pre seba alebo aby sa mohol pochváliť ostatným. Nájde si miesto, odkiaľ má celú v zábere, môže to byť socha, budova prípadne iný zaujímavý objekt. Určite sa nájdu ľudia, ktorí vedome alebo nevedome vstúpia do záberu. Môžu iba prejsť medzi fotografom a cieľom snímky. Potom sú aj takí ako on, ktorí si chcú odfotiť pamiatku. Ďalšia skupina ľudí sa iba kochá umeleckým dielom. Fotografovi zostáva iba čakať pokiaľ nežiaduce elementy zmiznú zo záberu. Niekedy to trvá chvíľu. Inokedy je to skoro nemožné. Jedno z riešení spočíva vo vytvorení viacerých fotiek a postupným skladaním vybrať časti, ktoré sú očistené od ostatných objektov. Týmto spôsobom, rekonštrukciou pozadia z viacerých fotografií, je možné získať aj kvalitnú výslednú fotografiu.

## 2 Fotografia

Je ťažké definovať dobrú fotografiu. Pre každého môže byť tá správna iná. Nebudeme sa zamýšľať nad umeleckou fotografiou. V takomto prípade využívajú autori svoje overené metódy. Zameriame sa na obyčajnú fotografiu. Získanú napríklad z dovolenky. V úvode bolo spomenuté ako problematické môže byť odfotenie budovy v centre mesta. Vysoká frekvencia ľudí, dopravných prostriedkov a prípadne zvierat naruša správnu kompozíciu. Tu sa ponúka myšlienka zostavenia požadovaného obrazu skladaním. Scénu odfoťíme niekoľkokrát a z každej snímky vyberieme to čo tam chceme mať. Za predpokladu, že je každá časť objektu aspoň na jednej fotografii. Často je zachytený objekt skoro celý okrem malej časti, na ktorej sa nachádza napríklad turista. Ak by sme chceli získať viac fotografií z rôznych miest, nastalo by stále dookola prácne a nudné nahrádzanie zlej časti obrazu. To by nebolo žiadané. Aj tu môže pomôcť počítač. Pretože sa jedná čiastočne o podobné a automatické úkony. Program analýzou fotografií zistí čo tam patrí a čo nie. Zo správnych častí obrazu vytvorí výslednú fotografiu, ktorá bude obsahovať len pozadie.

### 2.1 Pozadie

Odlíšiť pozadie od iných nechcených objektov je hlavný problém. V prípade dobrých vhodne získaných fotografií sa za pozadie dá považovať oblasť, ktorá je zachytená najčastejšie. Pretože auto, ktoré prechádza cez cestu pred nejakou budovou sa pohybuje a teda bude na každom zábere na inom mieste (to je ideálny prípad, nemyslíme autá trvalo parkujúce atď.). Považovanie najčastejších dát v obraze za pozadie je prvá možnosť. Iný spôsob získania pozadia je vyhľadávanie určitých objektov a ich následné odstránenie.

### 2.2 Vyhľadávanie určitých objektov v obraze

V súčasnosti existuje veľa rôznych programov slúžiacich na rozpoznávanie napríklad ľudí v obraze. Preto, ak máme na fotografii budovu a pred ňou sú nejakí ľudia, algoritmus ich vie detekovať. Zistí, že sú v určitej oblasti hľadané objekty, pozrie na inú vzorovú fotografiu na tú istú oblasť. Ak tam ľudí už nenájde, tak sa použije do výsledku táto oblasť. Takýmto postupným skladaním a vyhodnocovaním je tiež možné získať čisté pozadie. Hlavná nevýhoda je, že treba vedieť čo sa má považovať za narušiteľov. Určitý algoritmus vie nájsť ľudí. Dá sa tušiť, že takýto postup sa dá použiť v špecifických prípadoch. Napríklad pri fotení sochy v centre námestia sa dajú predpokladať turisti. Ak ich program zistí a nahradí, je dobrý výsledok hotový. Ak by sme použili hotový algoritmus na objekt, pred ktorým sa nachádzajú aj autá, tak s nimi sa program nevysporiada. Bude ich považovať za pozadie. Program sa môže doplniť o algoritmus hľadania áut v obraze. Potom by sa vyrovnal aj s týmito problémami. Určite sa potom nájde zase niečo nové. Napríklad pôjde človek so psom. A rôzne ďalšie nežiaduce objekty. Tento postup je dobrý iba pri špecifickom type záberov. Ak sa dá typ zaručiť potom je postup vhodný.

## 2.3 Použitie

So získavaním pozadia z obrazu sa už veľa pracovníkov alebo výskumníkov z rôznych oblastí stretlo. Len to nemusí byť na prvý pohľad jasné. A tiež podľa výsledných aplikácií je ťažko poznať akým spôsobom algoritmus pracuje.

Program Google Earth zobrazuje snímky zemského povrchu. Určite sa vývojári stretli s problémom ako odstrániť oblačnosť. Z družíc získali veľké množstvo dát. Problém bol určite pospájať snímky aby sa správne prekrývali. Na niektorých boli určite mraky zakrývajúce zemský povrch. Tie bolo treba odstrániť. Podobne sa dá predpokladať, že 3D budovy v tomto programe mali podobný problém. Zvlášť historické centrá veľkých miest je nemožno odfoťiť čisto bez nadbytočných objektov.

Získavanie pozadia sa môže použiť aj v určitých typoch reklamnej fotografie. Ak je treba získať záber určitej scény presne ako si želá autor. Väčšinou je však treba získať dokonalú fotku. V určitých prípadoch by takýto program mohol nájsť uplatnenie. Ak by boli fotografie rôzne alebo výsledky by neboli dostatočné, mohol by taký algoritmus aspoň pomôcť naznačiť, kde treba obrázok ručne upraviť.

Podobne ako reklamnú fotografiu môže byť problém získať dobovú fotografiu. Moderné autá a ľudia oblečení v dnešných šatách na fotografii stredovekého zámku sa určite pre niektoré účely nehodia.

Ďalšie uplatnenie sa môže objaviť pri tvorbe pohľadníc. No nie vždy je takýto výsledok žiadaný. Niektoré pohľadnice obsahujú turistov úmyselne, aby sa tým dalo najavo, že o dané miesto je záujem.

Rekonštrukcia pozadia by sa dala vzdialene použiť aj v bezpečnostných systémoch. Stačí ak sa bude snímať nejaká plocha v určitých časových intervaloch. Následne bude na získaných dátach aplikovaný program. Tentokrát nebude rekonštruovať pozadie ale iba sa o to pokúsi. Bude analyzovať či sa nejaká oblasť nelíši od tej istej na predchádzajúcich fotografiách.

Iné použitie môže byť vizuálna kontrola stavby. Za predpokladu, že sa robotníci budú pohybovať. Potom nie je problém zo série fotografií získať výslednú fotografiu budovy a vidieť čo z toho vzniká. Prípadne tieto fotografie použiť vo forme časozberného videa.

Podobne ako bezpečnostný systém je možnosť kontroly zmien aj inde. Niektoré činnosti v priemysle by to vedeli uplatniť. Fotografie získané v určitých intervaloch sa budú spracúvať. Následne algoritmus pri zostavovaní pozadia zistí, či na fotografiách tej istej scény nie sú odchýlky. Ich výskyt môže znamenať chybu. Napríklad porucha pásu by znamenala hromadenie produktov atď. To by sa zistilo. Takisto sa práve naopak môže kontrolovať, či sa zmena deje. Napríklad pravidelné kontroly činnosti robotov.

Je zrejmé, že veľká časť sa kryje s videom. Tá má oproti fotografii výhody a aj nevýhody. Nesporné plus je pohyblivý a plynulý obraz. Medzi nevýhody patria vyššie nároky na výkon a úschovu dát. Video zaberá oveľa viac pamäte. Preto je možné to video a fotografiu skombinovať. Napríklad video kamera bude získavať len jednotlivé fotografie. V prípade niečoho nezvyčajného sa spustí videozáznam a človek starajúci sa o danú činnosť bude môcť podľa neho určiť čo sa stalo.



### 3 Problémy pri rekonštrukcii pozadia

V ideálnom prípade by bol obraz tej istej scény úplne identický okrem narušiteľov. Tým sa myslia ľudia nachádzajúci sa v zábere. Obdobne zavadzajú dopravné prostriedky. Nesmie sa zabudnúť na zvieratá. Či už v doprovode ľudí alebo voľno pohybujúce sa zvieratá. Napríklad sú to vtáky lietajúce medzi zdrojom fotografie a fotografovým objektom. Taktiež problémy môžu spôsobiť holuby prechádzajúce sa po zemi. Tieto objekty sú nevhodné a preto sa ich snažíme odstrániť zo záberu. Okrem nich sú v snímkach problematické ďalšie veci, na ktoré by sa bez skúseností s danou problematikou mohlo ľahko zabudnúť. Sú to napríklad tieň. Nezáleží akého sú pôvodu. Ľudské sa vyskytujú s osobami a teda nepredpokladá sa ich stála účasť vo vzorových dátach. Naproti tomu tieň, ktoré vrhajú budovy sú horšie. Pokiaľ nie sú k dispozícii fotografie nasnímané počas celého dňa, určite sa vo výsledku nejakým prejaví. V prípadoch väčšieho časového úseku to môže byť aj tieň, ktorý sa trochu posúva a môže spôsobiť problémy s rozpoznávaním, či tam patrí ako pozadie alebo je to premenlivý objekt. Kvôli uvedeným problémom pomáha ak je cieľ fotený priamo a slnko je za zdrojom dát. Žiaľ nie vždy sú na to vhodné podmienky.

#### 3.1 Počasie

Významný vplyv na kvalitu má aj počasie. V prvom prípade ani nie je na fotkách priamo zachytené. Už samotné zvýšenie intenzity osvetlenia spôsobené vyjdením slnka spoza mrakov má nežiaduce efekty. Okrem spomenutých tieňov sa to prejaví ako tmavnutie alebo zosvetlenie fotografovaných objektov. V prípade častejšieho striedania svetelnosti to môže spôsobiť, že niekedy predmet vrhá tieň a niekedy nie. Následne na vodnej ploche, či už mláka alebo vodná nádrž, v častých prípadoch aj na mokrej vozovke vznikajú odrazy. Tento jav je dosť problematický. Veľakrát je tiež tento efekt zachytený na fotografiách úmyselne. Z umeleckého hľadiska niekedy môže fotografiu výrazne zlepšiť.

Vplyv počasia priamo sa nachádzajúceho v zdrojových dátach je mierne rozdielny. Veľké problémy spôsobuje dážď. Jeho odstránenie je veľmi náročné. Už letný pohľad na fotografiu zasiahnutú týmto neduhom zanecháva zlý dojem. Toto sa však riešiť nebude, pretože je to náročný problém. A pokiaľ nie sú vstupné dáta dostatočnej kvality, ťažko sa môže dostať slušný výsledok.

Problém tvorí aj obloha v zábere. Ak sa nachádza na fotografiách bezoblačné a jamé počasie, tak je to dobré. Horšia situácia nastane, ak je čiastočne alebo úplne zamračené. Podľa použitej metódy sa dá získať vo výsledku prirodzená a bez artefaktov obloha aj z horších dát. Často to však nedopadne najlepšie. Pretože vo veľkej skupine prípadov sa mraky vplyvom vetra posúvajú. Je ťažko detekovateľné, či je to ten istý oblak alebo iný. Súvisle zamračené je najhoršie. Obloha je rôzne svetlá a tmavá. Spôsobuje to prechod svetla cez nerovnako vysoké mraky. Vietor následne spôsobí, že na každej fotke je obloha úplne iná.

Vietor má negatívny vplyv aj na ostatné predmety. Môže to byť napríklad zástava na stožiaroch. V prípade silnejšieho vetra nielen spôsobí rozdielne zobrazenie na viacerých fotografiách, ale môže vytvoriť nejasné vykreslenie objektov. To môžu byť napríklad hýbajúce sa konáre stromov. Taktiež patria medzi zhoršujúce elementy. Najmä tenké vetvičky alebo lístie stále menia svoju polohu.

Veľký problém môže znamenať osvetlenie a jeho zmena. V interiéroch to až tak nehrozí. Horšie je to v exteriéroch. Spôsobuje to predovšetkým počasie. Záleží na fotoaparáte ako sa s tým vyrovná. V prípade uzamknutej expozície môže nastať preexponovanie alebo podexponovanie

snímku. Ak sú počítané expozície pri každom odfotení, bude výsledok vyzerat' lepšie, ale bude mať oproti ostatným rozdielne vlastnosti.

Výsledkom snaženia je eliminácia hore uvedených problémov. Komplikáciu spôsobujú rôzne javy. Čím sa ich v pôvodných dátach nachádza viac, tým je to horšie. Preto sa ťažko získavajú dobré výsledky zo zlých fotografií. Obsah rušivých elementov za premenlivého osvetlenia spôsobuje odrazy a tieňe.

## 4 Vybrané algoritmy

Cieľom výsledného programu je získať fotku, ktorá bude zbavená nežiaducich elementov. Na začiatku získame série viacerých fotiek. Predpokladá sa, že budú rovnakého rozlíšenia a typu. Je dôležité aby bolo snímané zo statívu alebo inej pevnej a nehybnej podložky. Ak by bola scéna čo i len kúsok posunutá bolo náročné zistiť či sa jedná o tú istú alebo inú. Samotná detekcia posunutia obsahu je zložitá a program ju neobsahuje. Následnou analýzou a prácou programu je rozlíšiť časti snímky, ktoré tvoria pozadie a sú nepohyblivé od pohyblivých. Tie sú nežiaduce. A na základe pozadia ich nahradiť, aby bolo pozadie kompletne.

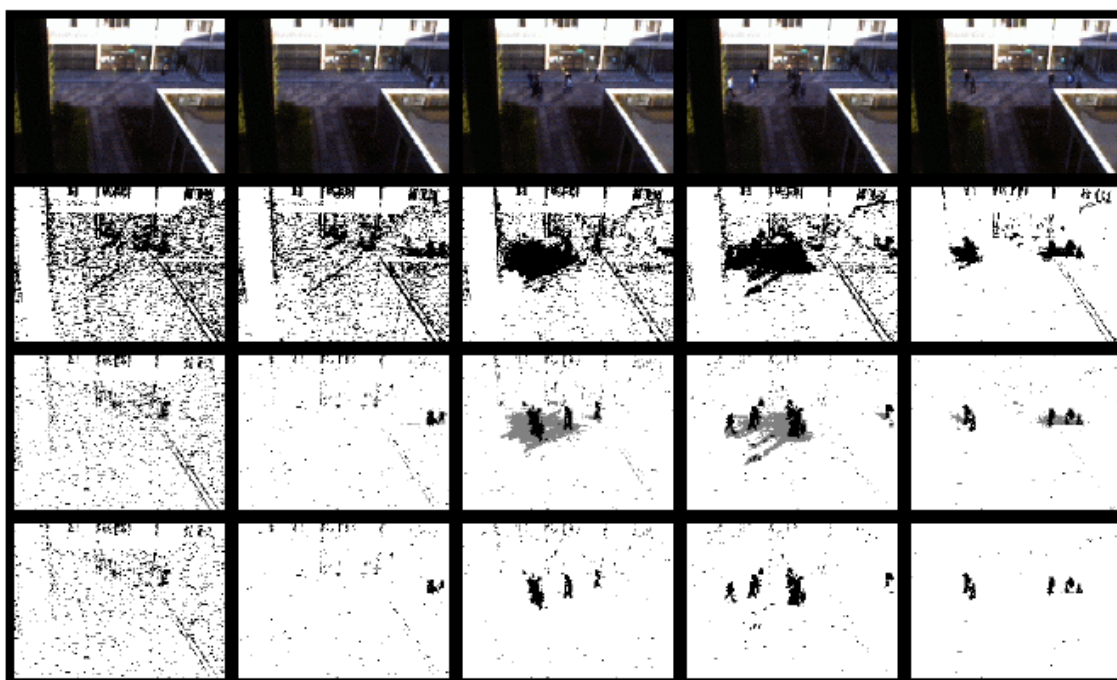
Je možné postupovať rôznymi spôsobmi. Najdôležitejšie je zistiť, čo do pozadia nepatrí. Rekonštrukcia obrazu sa vykoná následne vybratím časti pozadia na vzorovej fotke, na ktorej je požadovaná oblasť zachytená. Riešenia týchto problémov už boli čiastočne objavené. Niektoré algoritmy hľadajú určité charakteristické znaky. Prípadne opakujúcu sa oblasť obrazu, ktorá mení svoje miesto. Niektoré metódy sú menej vhodné pre fotografie. Sú určené pre video. Video neobsahuje tak do detailov zachytenú scénu ako samotná fotografia ale obsahuje neporovnateľne viac snímok. Nepredpokladá sa, že užívateľ bude mať k dispozícii stovky fotografií s rovnakou scénou. Ale na každom postupe sa dá nájsť aspoň princíp, ktorý sa dá po určitej obmene aplikovať aj na menej početnú vstupnú skupinu dát.

Postupov na zrušenie nežiaducich objektov je celkom dosť. Avšak horšie je to s ich uplatnením v praxi. Niektoré algoritmy vykazujú kvalitné výsledky iba pri určitých druhoch vstupných dát. Ďalej sú niektoré dosť zložité a vyžadujú aj veľké znalosti matematiky pre svoje pochopenie. Každý postup má svoje klady a zápory. Z môjho konkrétneho pohľadu za zápor považujem aj zásah od užívateľa. Výsledok mojej práce pracuje samostatne a neobsahuje žiadne užívateľské rozhranie. Práve preto by používateľova pomoc algoritmu bola len ťažko realizovateľná. A ak by bola tak by to bolo veľmi ťažkopádne. Tým však netreba hneď takéto postupy zavrhnúť. Ako bude uvedené ďalej niektoré metódy je možné sklbiť a užívateľov prístup vynechať. Častý neduh iných metód je iba ich uplatnenie vo videu. Pre video je podobný problém riešený už veľa krát. Niečo sa z toho dá aplikovať aj na fotografie. Ale nedosahuje to porovnateľné výsledky. Za najväčší problém je počet vzorových dát. Tieto algoritmy požadujú pre svoju činnosť stovky snímok statickej scény, čo v oblasti fotografii nie je možné dosiahnuť. Samozrejme aj postupy pre video môžu pomôcť s riešením niektorých typov problémov.

Už bolo rôznymi vedeckými pracovníkmi alebo odborníkmi vymyslených niekoľko algoritmov, ktoré sa týkajú problematiky rekonštrukcie pozadia. Z fotografií vedia získať pozadie, iné nájsť nežiaduce objekty. Prípadne sú schopné nájsť tie objekty a podľa tých zistiť či je objekt súčasťou pozadia alebo nie. Veľa algoritmov je navzájom si dosť podobných alebo sú vytvorené ako vylepšenie predchádzajúcich. Najzaujímavejšie algoritmy, ktoré už boli vymyslené by mohli slúžiť ako inšpirácia pre moju prácu. Sú to nasledovné: An improved adaptive background mixture model for realtime tracking with shadow detection a Object removal by exemplar based inpainting. Druhá menovaná je trochu odlišnejšia oproti prvej a aj oproti smeru akým sa moja práca ubera ale o to je zaujímavejšia. Algoritmus požaduje iba jednu vzorovú fotografiu. Avšak požaduje interakciu od užívateľa. Ten označí časť obrazu, ktorý nie je súčasťou pozadia. Program sa následne sám pokúsi vynechanú oblasť doplniť podľa okolia.

## 4.1 An improved adaptive background mixture model for realtime tracking with shadow detection

Algoritmus[1] slúži na nájdenie osôb a predmetov v obraze, ktoré nepatria nemennému pozadiu. Dalo by sa použiť na nájdenie nežiaducich predmetov na zdrojových fotografiách. Následne ich nahradiť pozadím z iných fotografií, ktoré majú na ich miestach pozadie. Metóda vzniká upravením Grimson et al framework[2] a jeho doplnením o zisťovanie tieňu obraze.



Obrázok 1. Obrázok znázorňuje v prvom riadku pôvodné snímky číslo 15,105,235,290,1200. Druhý riadok je získaný metódou Grimson et al. Tretí a štvrtý je získaný popisovanou metódou s tieňom a bez neho. Tieň je znázornený šedou farbou. Prevzaté z [1].

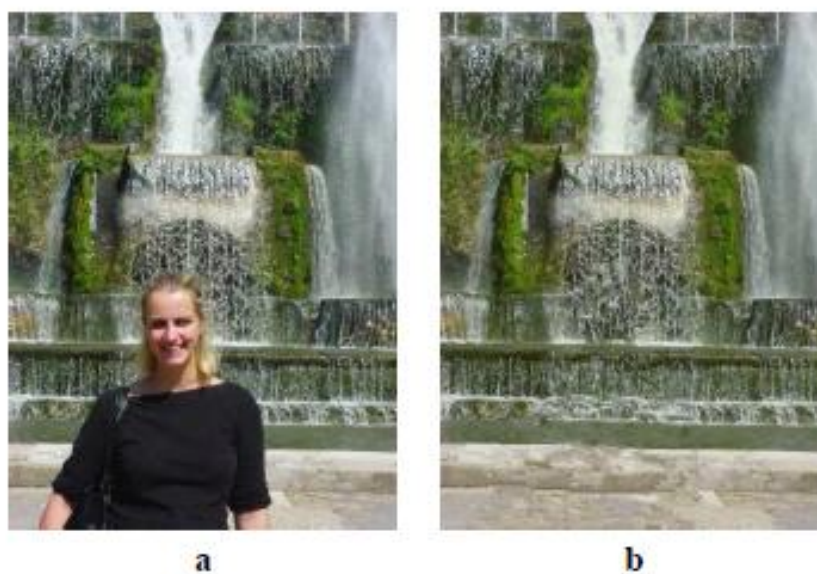
Využíva sa Adaptive Gaussian mixture model[3]. Každý pixel je modelovaný niekoľkými Gaussovými rozloženiami. Samotné rovnice pre výpočet veličín z predchádzajúcej správy sú doplnené a vylepšené. Zisťovanie tieňov je vzhľadom k pôvodnej metóde nové, pretože neexistuje heuristika, ktorá je schopná nájsť tieň v gaussových komponentoch. Na riešenie tohto problému bola použitá farebná škála obsahujúca chromatickú vlastnosť a zároveň aj jas. Potom sa porovnáva okolie predmetov nepatriacich pozadiu. Ak sú hodnoty farieb a jas v rozmedzí určitých hodnôt, sú pixely považované za tieň. Žiaľ metóda dosahuje úspešné rozpoznanie pozadia od nestálych predmetov až pri použití veľkého počtu fotografií ako možno vidieť na priloženom obrázku. Zaujímavé na tomto postupe je aj použitie farebnej škály obsahujúcej aj jas.

## 4.2 Object removal by exemplar based inpainting

Metóda[4] pracuje iba s jedným obrázkom. Užívateľ na ňom označí určitú oblasť. Vybraná časť sa vyplní podľa zbytku obrazu a hlavne podľa blízkeho okolia. Vizuálne obraz vyzerá veľmi vierohodne. Určite má tento postup budúcnosť. Avšak moja práca pracuje úplne automaticky a užívateľ nemá ako zadať oblasť pre upravenie. Možnosť aplikácie tejto metódy je, aby program sám zistil oblasť určenú na nahradenie. Následne na danú oblasť by bol použitý tento algoritmus. Týmto spôsobom som pracoval ale zatiaľ som nedosiahol požadované výsledky.

Jedná sa o spojenie dvoch metód. Texture synthesis[5] a inpainting[6]. Prvá nahrádza oblasti obrázku jednoduchými vzormi. Druhá tiež nahrádza časť obrazu. Opakuje dvojrozmerné vzory s určitou nepravidelnosťou. Výsledný algoritmus sa snaží použiť to najlepšie z nich. Veľmi dôležitý je postup akým jednotlivé kroky nasledujú za sebou.

Na začiatku sa určí veľkosť template window - autori navrhujú 9x9 pixlov. Každý bod má dve vlastnosti: farbu a confidence. Tá určuje akú dôveru máme k farbe pixlu. Samotný algoritmus začína vypočítaním confidence pre každú vzorku pixlov na hranici vybranej oblasti. Pomocou nej sa získa priorita pixlu. Nasledovne sa vyberie vzorka s najväčšou prioritou. Tá je šírená do vnútra oblasti. Potom sa zase počíta confidence. A tak až pokiaľ nie je doplnená celá oblasť.



Obrázok 2. Ukážka práce algoritmu. Pôvodný obrázok je a. Osoba bola manuálne označená. Na obrázku b bola dopočítaná. Prevzaté z [4].

Ilustračné obrázky demonštrujú silu tohto algoritmu. Uplatnenie v mojej práci viazne na úspešnej detekcie oblasti. Ak by sa dalo spoľahlivo a presne zistiť kde je objekt, ktorý do obrázku nepatrí už by nebola prekážka proti použitiu tejto metódy.

## 5 Implementácia

Výsledný program je vo forme konzolovej aplikácie. Pri spustení očakáva parametre názvy súborov. Sú to názvy zdrojových fotografií. Pokiaľ už nebude uvedené inak, program vytvorí jednu fotografiu. Aplikácia využíva viacero spôsobov ako získať požadovaný výsledok.

### 5.1 Pamäťové štruktúry

Každá fotografia je v programe reprezentovaná poľom štruktúr. Tie sú alokované dynamicky po spustení a po skončení výpočtov bude pamäť korektne uvoľnená. Štruktúra pozostáva zo štyroch unsigned char. Každý reprezentuje jednu farbu z RGB farebného modulu. Posledný unsigned char obsahuje priehľadnosť – alpha kanál. Ten však nie je používaný, pretože program pracuje s formátmi JPG a BMP. Tie priehľadnosť neumožňujú. BMP je zvolený pre svoju relatívne jednoduchú štruktúru. Obsahuje hlavičku a potom hodnoty farieb každého bodu. Jeho načítanie a ukladanie preto nie je zložité. Nevýhoda je veľkosť obrázku. Práve preto je ďalší použitý formát JPG. Používa kompresiu aby nezaberal toľko veľa pamäte. Daň za túto vlastnosť je zložitá práca s obrazovými dátami. Oproti BMP je častejšie používaný. Na internete a tiež v digitálnych fotoaparátach sa teší obľube. Na prácu s ním je v práci použitá knižnica funkcií FreeImage. Jedná sa o open source projekt a preto je možné v prípade záujmu naštudovať si jej zdrojový kód. Obsahuje funkcie na prácu s veľkým počtom rôznych grafických formátov. Vďaka tomu je vhodný aj pre možné rozšírenie v budúcnosti. Samotná štruktúra uchováajúca farebné hodnoty bodov je tiež pripravená pre rôzne typy súborov, pretože prítomnosť alpha zložky umožňuje používať formáty podporujúce priehľadnosť ako PNG alebo GIF. Pre každý bod fotografie je uložená jeho farba. Z toho vyplýva, že pamäťová náročnosť programu závisí od rozlíšenia a počtu zdrojových fotografií. Okrem dát fotografií sa používa ešte niekoľko pomocných dátových štruktúr slúžiacich na medzivýpočty.

Program je možné používať v operačnom systéme Windows a tiež v Linuxe. Verzia pre Windows sa prekladá v prostredí Microsoft Visual Studio (vytvorené verziou 2008). Pri používaní programu je nutné aby v priečinku s aplikáciou bola umiestnená aj knižnica FreeImage.dll. Slúži na načítavanie a ukladanie fotografií vo formáte JPG. V prostredí Linux (pri použití g++ 4.3.1) je nutné mať FreeImage nainštalované už pred prekladom samotného programu. Následne však už program funguje samostatne a nie je potreba, aby bol v rovnakom priečinku ešte nejaký iný súbor (ako vo Windows knižnica dll).

Samotné načítanie vzorových dát prebieha dvomi spôsobmi. Ak sú dáta vo formáte BMP tak sa z hlavičky prvého súboru zistia rozmery obrázku. Na ich základe sa alokuje potrebná pamäť. Následne sa dáta uložia do na to určených štruktúr. V prípade formátu JPG sa načítajú obrázky pomocou FreeImage knižnice do ich dátových typov. Funkcia na zistenie rozmeru obrázku získa rozmery a alokuje pamäť. Potom sú dáta skopírované do rovnakých štruktúr ako pri BMP. FreeImage uvoľní pamäť všetkých obrázkov okrem prvého. Do neho sa spätne uloží výsledok aplikácie a potom prevedie z internej interpretácie do JPG a vytvorí výsledný obrázok. Následne sa uvoľní jeho pamäťové miesto.

## 5.2 Funkcie rekonštrukcie pozadia

Ak už máme dáta úspešne prístupné v pamäti programu, možne postupne volať funkcie na ich spracovanie. Samotných funkcií je viacero druhov. Každá funkcia dosahuje najlepšie výsledky pri určitých vlastnostiach fotografií. Napríklad niektoré dosahujú dobrý výsledok pri menšom počte zdrojových dát a iné pri väčšom. Takisto niektoré sa lepšie vyrovnávajú s rôznou intenzitou osvetlenia ako iné. Úspešnosť funkcie na kvalite získania čistého pozadia závisí aj od počtu vzorových fotografií. V nasledovnom texte budú použité neurčité termíny slúžiace na približný popis počtu vstupných dát. Slovné spojenie málo vzorových fotografií sa myslí počet menší ako 10. Bežný počet býva v rozmedzí 10-20. A ak je dodaných fotografií viac ako dvadsať, bude uvedené slovné spojenie väčší počet fotografií. Avšak toto neznamená, že ak určitá metóda dosahuje dobré výsledky pri menšom počte fotografií, pri väčšom počte je nepoužiteľná.

Nasleduje podrobnejší popis mnou vytvorených funkcií a vyzdvihnutie ich kladných ale aj záporných vlastností. Výsledné obrázky sú získané zo vzorových fotografií uvedených v kapitole testovanie. Pri popise úspešnosti funkcií je spomínané slovo duch. Myslí sa tým objekt čiastočne priesvitný. Vznikne nedostatočným odstránením daného objektu. Najčastejšie variáciou priemerovania.

### 5.2.1 Priemer

Môže sa považovať za úplne najjednoduchšiu metódu získania pozadia. Pre každý bod fotografie sa vypočíta jeho hodnota aritmetického priemeru. Počíta sa pre každú farbu samostatne. Podľa očakávania výsledky nie sú najlepšie. Pre časť obrazu, ktorá je nemenná, je výsledok výborný. Horšie je to s časťami obrazu, kde sa nachádza raz pozadie a inokedy nežiaduci objekt. Na tomto mieste sa vo finálnej fotke objavia škvrny alebo duchovia. Táto metóda bola skúšobná, aby sa dalo zo začiatku s čím porovnávať. Samostatne do finálnej verzie programu začlenená nie je. Avšak niektoré iné metódy, pokiaľ nedosiahnu požadovaný výsledok, určité body získajú týmto spôsobom.



Obrázok 3. Výsledná fotografia získaná funkciou priemer. V pravej časti je výrez fotografie, na ktorom je vidieť duchov po električke a ľudoch.

### 5.2.2 Rozmedzie

Funkcia, ktorá je najvyužívanejšia pri výpočtoch. Na svoju prácu požaduje parametre. Prvý slúži ako adresa výstupného obrázku. Druhý a tretí sú dôležité pre samotný algoritmus. Druhý určuje



minimálny počet zhôd a tretí toleranciu farebných odtieňov. Funkcia prechádza každý bod obrazu. Načíta hodnoty bodu na prvej fotke. V pomocných premenných sú tieto hodnoty zväčšené a zmenšené podľa veľkosti tolerancie. Následne sa vyberie ten istý bod z ďalšej fotografie. Skúma sa, či hodnoty tohto bodu sa nachádzajú v rozmedzí hodnôt vzorového bodu. Ak áno, tak sa inkrementuje počítač zhôd. Cieľom je získať viac zhôd ako  $1/\text{parameter\_funkcie\_minimálny\_počet\_zhôd}$ . Ak sa to nepodariť, tak vzorový je použitý bod z ďalšieho obrázku. Pokiaľ by sa nezistil minimálny počet zhôd, tak sa hodnota bodu vo výsledku získa aritmetickým priemerom hodnôt bodu vo všetkých vstupných obrázkoch. Algoritmus nie je veľmi zložitý. Napriek tomu dosahuje veľmi dobré výsledky. Celkom dobre sa dokáže vysporiadať so zmenou intenzity osvetlenia. Tiež výsledky získané z malého počtu vstupov sú kvalitné. Ak je vstup veľký počet fotiek, ktoré sa líšia intenzitou svetla a aj častým výskytom objektov, ktoré chceme odstrániť, tak výsledok nie je najlepší. Ale dá sa zlepšiť na relatívne dobrú úroveň. Ale vyžaduje to postup pokus-omyl. Výsledný obrázok sa môže v závislosti na parametroch funkcie dosť líšiť. Menší počet dát najčastejšie poskytuje uspokojivé výsledky s parametrami 2/10 (parameter zhody/parameter okolia). Podobne je na tom 2/20 a 2/30. V určitých prípadoch vynikne aj nastavenie 3/10 a 3/20. Ostatné nastavenia nedosahujú dobrého výsledku. To platilo o malej vzorke vstupných dát. Pri jej náraste sa začínajú osvedčovať vyššie hodnoty parametrov. A tie nižšie už nedosahujú tak čistý obraz. Pre ľudské oko nastavenia 3/20 a až 4/20 vyzerajú najlepšie. Samotné testovanie bolo robené s rôznymi hodnotami parametrov. Začínalo sa 2/1. Ešte lepšie by bolo 1/1, ale to by okrem prípadov identického bodu vždy rátať priemer. A to samozrejme nie je účel tejto metódy. Na priložených obrázkoch je vidieť výsledky metódy pri rozumne zvolených parametroch. Podrobnejšie porovnanie výsledkov metódy a jej použitia s rôznymi parametrami sa nachádza v kapitole testovanie.

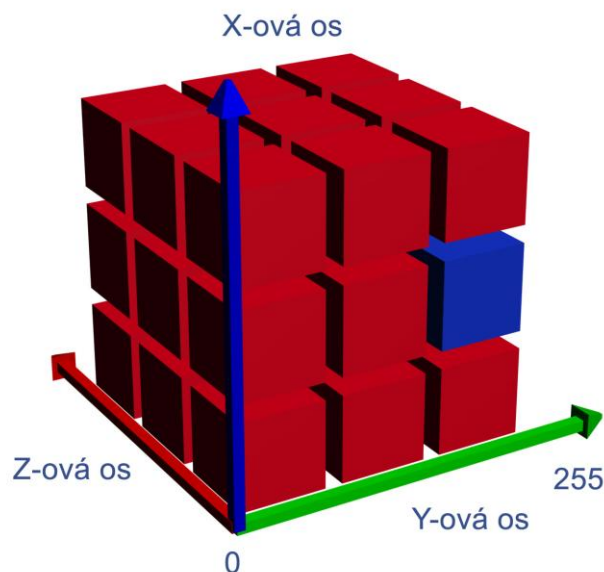


Obrázok 4. Fotografia získaná s parametrami 2/10. Vpravo je výrez demonštrujúci vyľudnenie zastávky.

### 5.2.3 Kocka

Tento spôsob výpočtu pracuje s kategóriami farieb. Možno si to predstaviť ako priestorový útvar kocka. Tú rozdelíme na 8 rovnakých častí. To dosiahneme tak, že na každej strane nájdeme body, ktoré sa nachádzajú v jej polovici. Ako osi kocky sa použijú farby. Teda x-ová os bude červená farba, y-ová os zelená a z-ová bude modrá. Pri rozdelení dĺžok strán kocky na polovicu získame 8 menších kociek, z ktorých sa skladá pôvodná veľká kocka. Pri použití 8 častí boli hodnoty dĺžky strán malých kociek rovné 127 (farebná zložka má rozsah 0-255 a polovica z toho je 127).





**Obrázok 5. Rozdelenie kocky na 27 častí. Každá os predstavuje jednu farebnú zložku RGB.**

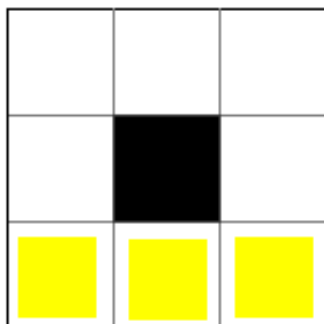
Každý bod obrázku používa jednu kocku. Podľa farebných hodnôt bodu sa zaradí do jednej malej časti. Nasledovne sa nájde časť, ktorá je najpočetnejšia. Z nej sa urobí priemerná hodnota a tá sa použije ako výsledok. Ako vybrať najlepšiu vzorku z malých častí je viac spôsobov. Zatiaľ sa osvedčil ako najlepší spôsob obyčajný priemer a priemer, ktorý vynecháva najkrajnejšie body. Samozrejme sa dá predpokladať, že táto metóda požaduje pre svoju činnosť viac zdrojových fotografií. Pri malom počte sa môže stať (aj sa to často vyskytlo), že časti obrazu budú tmavšie ako iné, prípadne sa vyskytuje výrazné zrnko. Preto okrem priemerovania s vynechanými najkrajnejšími bodmi je rovnako implementované priemerovanie zo všetkých hodnôt. Samozrejme, krajší výstup závisí od zdrojových dát. Takisto je možné použiť väčšiu kategorizáciu fotografií. V programe je použitá aj kocka s 27 časťami. Vtedy sa strany rozdelia na tretiny. Rozmer malej strany malej kocky je 85. Dá sa očakávať, že toto jemnejšie delenie by malo vykazovať lepšie výsledky. No nie je to úplne pravda. Výsledky pri 8 a 27 častiach sú dosť podobné a dosahujú rovnaké aj pekné a aj horšie oblasti. Rozdiely sú minimálne a ani pri použití väčšieho počtu vzorových fotografií, nie sú výsledky oveľa odlišné. Najvhodnejšie vzorky dát pre túto metódu tvoria fotografie, ktoré sú približne rovnaké. A teda je lepšie ak nie sú zahrnuté fotografie, na ktorých je aj veľa objektov určených na odstránenie. Na základe tohto poznatku je táto metóda (jedná sa o obe verzie-8 aj 27 častí) použitá na finálne úpravy. Najprv sa aplikuje na vzorové dáta iný spôsob, najčastejšie rozmedzie, a následne výsledky sú predané kocke. Vtedy sa tam neobjavia žiadne nechcené artefakty vo forme duchov a zrna ale vo väčšine prípadov sa obrázok ešte o kúsok (často výrazne) zlepši.



Obrázok 6. Fotografia získaná funkciou kocka. Vo výreze fotografie je vidieť odstránenie ľudí ale aj susediace oblasti s odlišným farebným odtieňom.

## 5.2.4 Susedia

Metóda, ktorá by mala mať najlepšie výsledky. Na základe výsledných fotografií sa však nedá povedať, že prevyšovala kvalitou ostatné algoritmy. Hlavnú myšlienku tvorí farba okolia. Jeden pixel sám o sebe nesie veľmi málo z pohľadu celého obrazu. Hocijaký útvar, pokiaľ nie je mikroskopických rozmerov, tvorí skupina bodov. Preto sa dá predpokladať, že susedné body majú rovnaké alebo podobné hodnoty. Samozrejme, že to neplatí stopercentne. Môžu byť rôzne hrany alebo rýchle farebné prechody. Ale ak zanedbáme tieto prípady, tak môžeme napríklad podľa blízkeho okolia bodu predpokladať jeho farbu. Prípadne, ak by bol výrazný rozdiel oproti nemu a ostatných bodov okolo neho, dá sa považovať za chybu v obraze. Napríklad ak je jeden bod sporný a okolie je rovnaké u viacerých obrázkov, môže bod zdediť farbu najbližších pixlov. Môže sa vybrať vždy nejaká najbližšie hodnota. Napríklad vždy ľavý alebo spodný bod (predpokladá sa postupný prechod bodov z ľavého dolného rohu obrazovky). Prípadne sa vyberie najčastejšia hodnota z najbližších bodov v okolí. Podobné výsledky dosahuje aj aritmetický priemer najbližších bodov. Základ je, že bod má určitý vzťah k okolitým pixlom. A preto ak sa z nejakého hľadiska zistí, že jeho hodnota nie je správna, môže sa upraviť. A úprava je najdôležitejšia. Ponúka sa niekoľko rôznych spôsobov ako to vyriešiť. Niektoré sú už načrtnuté vyššie. Zaujímavé výsledky dosahujú aj ich kombinácie. Napríklad sa jedná o priemer hodnôt bodov nachádzajúcich sa pod skúmaným bodom. Priemer najčastejších hodnôt okolia a ich vplyv na pixel sa získa tak, že sa spriemerujú s daným bodom váženým priemerom.



**Obrázok 7. Čierna znázorňuje aktuálne počítaný bod. Môže sa použiť celé okolie bodu alebo napríklad iba predchádzajúci riadok označený žltou farbou.**

Moja implementácia spočíva vo využití predchádzajúcich metód. Na začiatku sa prejde celý obrázok bod po bode. Vyberie sa najčastejšia hodnota podľa funkcie rozmedzie. Nie je jednoduché vybrať najvhodnejšie parametre rozmedzia. Ale dá sa povedať, že hodnoty zistené pokusmi pri nej sú vhodné aj pre toto použitie. Body, ktoré vyhovujú rozmedziu sú označené ako správne a s nimi sa už manipulovať nebude. Slúžia ako vzory pre svoje okolie. Následne sa pôjde tiež bod po bode a hľadajú sa body, ktoré rozmedzím neprešli a sú teda neurčité. Cieľom je získanie ich hodnôt na základe správnych bodov v ich okolí. V prípade, že vyhovujú nejakej (je možnosť použitia viacerých spôsobov ako už bolo spomenuté) metóde ich dopočítania tak sú označené za správne a slúžia ako vzory pre ďalšie body. Takýmto spôsobom sa ide až pokiaľ už nebudú všetky možné body dopočítané. Zostanú body, ktoré nevyhoveli ani rozmedziu ani tejto metóde. Dopocítavanie nerozhodných bodov prebieha viacerými krokmi. Najprv sa zistí či má bod okolo seba 9 bodov. Tomu nevyhovujú body nachádzajúce sa na okraji fotografie. Všetky rohy a tiež všetky okraje. Tieto body sú počítané extra. Zistí sa koľko z okolitých bodov je podobných. Princíp je podobný rozmedziu. Ak je počet podobných bodov väčší ako medza, tak počítaný bod získa hodnotu jedného z nich. Týmto spôsobom sa program chová až kým nezostane obrázok hotový alebo body sa nevedia dopočítať. Túto časť je tiež možné vykonať viacerými možnosťami. Napríklad bod by mohol byť počítaný ako priemerná hodnota podobných pixlov okolia. Tiež by mohli byť získané ako priemer len bodov nad daným bodom alebo len pod ním. Žiaden postup z uvedených však nedosahuje ideálne výsledky. Preto táto metóda poskytuje dobré výsledky iba niekedy a na dokonalé získanie obrazu treba objaviť správny algoritmus nahrádzania týchto bodov. Naimplementované riešenie dosahuje relatívne slušné výsledky pri stredne veľkom počte vzorových dát. Väčší počet vzorových fotografií alebo ich veľké rozmery sú pre túto funkciu nepriaznivé. Je príliš časovo náročná. Pri väčšom množstve fotografií sa v určitých prípadoch výsledok zlepši alebo zhorší. Pri rôznorodejšej povahe vstupných dát táto metóda má problémy. Buď zostanú v obraze body, ktoré sa nevyrátajú alebo sa s týmito bodmi algoritmus vyrovná priemerovaním. Ale to väčšinou nie je ideálne. Obzvlášť ak je obraz dosť premenlivý. Body nezískajú správnu hodnotu. A postupným distribuovaním sa chyba roznesie ďalej. A potom vzniknú machule alebo duchovia v obraze, čo sa za žiadaný výsledok považovať nedá.



Obrázok 8. Funkcia susedia a jej výrez zobrazujúci zastávku.

Podľa rôznorodosti vzorových dát, ich zložitosti a samotnej susedskej metóde tvoria nerozhodné body určitú časť obrazu.

Ideálne by bolo sa zbaviť aj toho zbytku. Avšak ak by sa použil spôsob, ktorý opraví všetky body tak obrázok nevyzerá vôbec pekne. Zatiaľ nebol použitý žiadny algoritmus, ktorý by bol natoľko vhodný aby vytvoril pekný výsledok a využil všetky body. Tu sa práve naskytá možnosť použiť spôsob vynájdený niekým iným. Konkrétne už skôr spomenutý Object removal by exemplar based inpainting . Na začiatku by sa dáta prehrali štandardne cez susedov s nejakými parametrami. Tie by boli nastavené tak, aby sa použila väčšia časť obrázku. Zostala by neurčitá oblasť. Samozrejme by sa to mohla vyskytnúť viackrát nezávisle na sebe. Predpokladá sa však, že značné okolie oblasti by boli určené už susedmi a jeho body by teda boli označené ako správne. Následne by sa obraz prešiel ešte raz. Teraz by sa hľadali body, ktoré sú hraničné pre správne a neurčité body. Týmto bodom by sa nastavila vlastnosť hraničných bodov. Na základe správnej časti obrazu, zlej a hraníc by sa mohla zavolať metóda Object removal by exemplar based inpainting. Pri správne vybraných oblastiach by výsledok mohol dopadnúť celkom dobre. Problém je správne vybrať oblasti, aby ich bolo málo a boli väčšie. Zatiaľ sa skôr darí vyprodukovať veľa malých plôch, ktoré sú blízko seba. Preto som zatiaľ nevyužil ďalšiu metódu na finálnu úpravu. Ale ak by sa podarilo susedov zlepšiť na dostatočnú úroveň určite by to stálo za to.

## 5.3 Výsledný postup

Postupným skúšaním rôznych metód a ich kombináciou bol stanovený postup, ktorý sa prednastavene použije. Načítajú sa vstupné dáta a uložia sa následne do pamäťových štruktúr programu. Následne sa zavolá metóda rozmedzie. Konkrétne s týmito parametrami: 2/10, 2/20, 2/30, 3/10, 3/20. Tým sa dosiahnu medzivýsledky. Na ne je aplikovaná metóda kocka. Požité nastavenie je 8 častí a priemer so všetkými hodnotami. 27 častí nespôsobí veľký rozdiel. Ak by sa použil priemer bez krajných bodov tak, by hrozilo, že obraz bude mať rozdielne susedské oblasti. Spôsobilo by ich veľká zmena jasu a zvýraznené niektoré farby. A ako posledná funkcia sa zavolá kocka. Vytvorí výsledný obrázok a program skončí.

Program pre prednastavený postup riešenia sa spúšťa s parametrom „p“. Okrem toho je možné spustiť ho aj s parametrom „v“. V tomto prípade sa nevykoná vyššie uvedený prednastavený postup. Budú vypočítané obrázky všetkými metódami. Každá metóda sa vykoná s určenými parametrami iba raz a výsledok sa uloží tak, že názov výstupného obrázku bude metóda a jej parametre. Takto vznikne 22 obrázkov. Rozmedzie vytvorí 18 výstupov s parametrami od 2/5 až 2/40



a takto isto aj pre  $3/5$  až  $3/40$  a to isté aj pre  $4/5$  až  $4/40$ . Kocka vytvorí 4 výstupy. Dvakrát aj s priemerovaním krajných bodov a dva razy bez nich.

Tento postup slúži v prípade, že prednastavený výsledok nespĺňa užívateľovu predstavu. Takto mu program vyprodukuje výsledky všetkých metód. Následne užívateľ na obrázky pozrie a zistí, či každá metóda vracia zlé výsledky alebo iba niektoré metódy majú problémy. Potom si môže nechať tie dobré obrázky. Prípadne tie dobré môže použiť ako vstup programu. Ak užívateľ spustí program so vstupmi, ktoré on považuje za dobré (podobajú sa v určitých veciach a už sú čiastočne spracované), šanca na kvalitný výsledok sa zvyšuje. Posledná možnosť je spustiť program s parametrom „s“. Vtedy sa použije iba funkcia susedia. S parametrami 2/20/5, 2/20/6 a 2/20/7. Samostatné spustenie týchto funkcií je z dôvodu ich časovej náročnosti. Nedosahujú až tak kvalitné výsledky, aby sa oplátilo na ich počítanie čakať.



Obrázok 9. Výsledná fotografia získaná s predvoleným nastavením.

## 6 Testovanie

Pri tvorbe práce bolo treba sledovať ako úspešne algoritmy pracujú. V prípade nejakej zmeny prebiehali testy stále dokola aby sa zistili či výsledok zlepšujú alebo zhoršujú. Tiež sa podľa výslednej fotografie dalo povedať, ktoré algoritmy sa môžu zlepšiť alebo ich treba prestať používať. Základné testovanie prebiehalo na sade obrázkov autobusovej a zároveň električkovej zástavky. Sadu 16 fotografií je možno vidieť na obrázku. Sú to zmenšené náhľady. Je na nich vidieť aké objekty má algoritmus za úlohy odstrániť a upraviť.



Obrázok 10. Sada vzorových fotografií.

Hlavnou úlohou je odstránenie autobusov a električiek, nachádzajúcich sa v pozadí fotografie. Tie zaberajú vždy veľkú časť fotografie. A tiež sú v strede, takže sú dobre viditeľné. Potom sú ešte zachytení ľudia, ktorí čakajú na svoj spoj alebo z nejakého práve vystupujú. Ďalšie priložené fotografie reprezentujú výsledky práce. Pohybujúca obloha narobila komplikácie každému algoritmu. Pretože mraky sa priebežne posúvali a to znamená, že farebná hodnota bodu bola vždy iná. Jasnejšie časti sa striedali s tmavšími a to spôsobovalo, že ani jedno nie je pozadie ale vždy iba prechodný objekt. Najčastejšie vznikali viditeľné oblasti s podobnými farbami, ktoré ale navzájom boli na prvý pohľad odlišné. Budova za zástavkou je nemenná celý čas a preto nespôsobovala problémy. Autobusová zastávka bolo prázdna, obsadená čakajúcimi ľuďmi alebo zakrytá pristaveným autobusom. Vo výslednej fotografii táto oblasť väčšinou zostala vyčistená. Miestami sa objavovali duchovia čakajúcich ľudí. Hlavne sa jedná o lavičku pod prístreškom. Skoro vždy sa tam niekto nachádzal. A ak po odchode čakajúceho si tam sadol niekto ďalší. Ľudia sa tam nachádzali na veľkom počte fotografií a algoritmy ich často brali už ako súčasť pozadia. Tento istý problém sa opakoval v prístrešku pred električkovou traťou. Tu výsledok dopadol často ešte horšie, pretože ľudia sa tu nachádzali na väčšine fotiek. Iba sa mierne posunuli, ale stále boli súčasťou obrazu. A tým vznikajú rozmazané osoby alebo duchovia osôb. Električky sa odstránili z fotografií bez vážnejších komplikácií. Opäť problematickú oblasť tvoril prístrešok s čakajúcimi ľuďmi. Pretože pri zastavení električky bol prístrešok schovaný za ňou. A táto oblasť bola najhoršia. Ani jeden algoritmus to nedokázal upraviť ideálne. Veľkú oblasť fotografie tvoria chodníky. Ľudí nachádzajúcich sa na nich sa podarilo väčšine algoritmov odstrániť. Výnimočne sa niekde vo výslednej fotografii nachádzajú postavy, prípadne ich tieň. Nakoniec ešte treba spomenúť zmenu osvetlenia. Prechodom oblačnosti s rôznou intenzitou vznikali rozdielne tmavé fotografie. Tento jav bol pre niektoré algoritmy nepriaznivý. Spôsoboval rôzne mapy zobrazením svetlejších a tmavších miest vedľa seba. Dopomohli tomu aj tieň, ktoré menili svoju intenzitu podľa jasnosti oblohy. Konkrétne výsledky s krátkym popisom sa nachádzajú pri každej funkcii. Časová náročnosť jednotlivých funkcií je pri počte 16 vzorových fotografií s rozmermi 3888 x 2592 na počítači s 2GHz procesorom a 2GB RAM rádovo v desiatkach sekúnd alebo v jednotkách minút. Okrem funkcie susedia, ktorá trvá výrazne dlhšie.



## 6.1 Rozmedzie



Obrázok 11. Rozmedzie 2/5.

Funkcia rozmedzie s parametrami 2/10 vytvorí peknú oblohu. Je to dané hlavne častým využitím priemerovania kvôli rozdielnym farebným hodnotám. Ľudí sa nepodarilo odstrániť. Opäť nebolo nájdených dostatočný počet zhôd a uplatnil sa priemer. To isté platí pre dopravné prostriedky.

Pri zmene parametrov na 2/20 je vidieť celkové zlepšenie. Jedine obloha sa zhoršila. Tak sa objavili rozdielne oblasti. Algoritmus detekoval viacero zhôd ale iba v blízkych oblastiach. Ale ľudia a električky sú na fotografii lepšie odstránení. Menší je aj počet duchov. Najväčší problém tvorí prístrešok v pravej časti fotografie.

Použitie parametrov 2/30 je v niektorých aspektoch lepšie a v iných horšie ako predchádzajúce nastavenie. Zhoršenie znovu nastalo v oblohe. Oblasť znateľne odlišná od susedných sa tu vyskytujú menej ale zato sú viac rozdielne. Ľudia zmizli z fotografie a tiež aj ich tieň. Aj po dopravných prostriedkoch nezostalo veľa artefaktov. Zhoršenie nastalo pri vždy problematickom prístrešku v pravej časti fotografie. A taktiež je viditeľné zhoršenie pri stretávaní trávnatých plôch a chodníkov. Vznikajú tu rozdielne oblasti ako pri oblohe.

Hodnoty 2/40 pokračujú v trende. Vznik susedných oblastí razantnejšie sa navzájom odlišujúcich pribúda. Vyskytujú sa opäť duchovia po ľuďoch, ktoré predchádzajúce nastavenie úspešne zmenilo. Použitie prvej hodnoty 3 namiesto 2 vykazuje podobné výsledky. Len kvalitná výsledná fotografia je v menšom rozpätí nastavení. Pri použití hodnoty 2 sa za najlepšie výsledky dá považovať 2/10, 2/20 a ešte aj 2/30 v niektorých prípadoch. Pri hodnote 3 je to iba 3/10 a občas 3/20. A pokusne použitá hodnota 4 nám vytvorí peknú fotografiu pri 4/10 ale aj to musia byť vhodné vzorové fotografie.

Na základe týchto pokusov bolo zistené, že najlepšie sú výsledky 2/10, 2/20, 2/30, 3/10, 3/20. A preto sú použité v predvolenom postupe tvorby výslednej fotografie.





Obrázok 12. Rozmedzie s parametrami 2/10.



Obrázok 13. Rozmedzie s parametrami 2/30.

## 6.2 Kocka

Na priložených fotografiách vyprodukovaných funkciou kocka je vidieť s čím si funkcia poradila s čím nie. Nasledovné fotografie sú získané kockou s 8 časťami.



**Obrázok 14. Fotografia získaná funkciou kocka. Priemer sa použil aj s krajnými bodmi.**

Fotografiu tvorí veľký počet oblastí s rozdielnymi odtieňmi farieb. Algoritmus sa nevie vysporiadať so vzorovými fotografiami, ktoré sa viac líšia. Ja prvý pohľad je vidieť fľakatú oblohu. Ale aj ľudia a dopravné prostriedky nie sú odstránené ideálne. Zostávajú duchovia. A na chodníku je vidieť aj výrazne bledšie miesta. Je to spôsobené tieňmi ľudí.



**Obrázok 15. Funkcia kocka bez použitia krajných bodov.**



Na ďalšej fotografii je fotografia počítaná bez krajných bodov. To výsledok dosť zmenilo. Obloha je krajšia. Je celistvá bez výraznejších rozdielov. Pri podobných farbách nepoužitie krajných hodnôt má priaznivé výsledky. Zbytok fotografie je podobný ako pri použití všetkých hodnôt. Výskyt duchov a výrazne odlišných oblastí je zjavný. V prípade použitia kocky s 27 časťami sú výsledky porovnateľné. Ani jeden výsledok nie je pekný. Preto sa tieto funkcie uplatňujú na fotografiách získaných inými metódami. V predvolenom postupe práce sa použije kocka po rozmedzí. Fotografia takto získaná sa použije ako výsledok. Na základe priloženého obrázku 12 z kapitoly implementácia je vidieť lepšie získanie pozadia. Obloha je skoro celá celistvá. Ľudia sú odstránení aj s dopravnými prostriedkami. Duchovia sa tiež až na pár výnimiek na fotografii nenachádzajú. Výnimka je časť obrazu niekoľkokrát spomínaná. Momentálne nie je implementovaný žiaden algoritmus schopný odstrániť objekty nachádzajúce sa na viac ako polovici vzorových fotografiách. Algoritmus ich považuje za pozadie. Za úspech sa v tomto prípade považuje aj ich rozmazanie, prípadne zmiznutie ich časti.

## 6.3 Susedia

Funkcia susedia dosahuje slabšie výsledky. Obloha je skoro vždy nekvalitná. Niektoré jej oblasti sú výrazne odlišné ako iné. Pri použití extrémnych parametrov sa dá získať pekná obloha ale zbytok fotografie už bude zlý. Ľudí sa podarí odstrániť čiastočne. Miesta s nevelkým výskytom nežiaducich objektov sa zrekonštruujú úspešne. Ale časti fotografií kde je veľký výskyt ľudí a dopravných prostriedkov nedosahujú žiadajú kvalitu. V lepších prípadoch sa vyskytujú duchovia a v horších časti týchto objektov. Tiež je možno vidieť na výsledných fotografiách oblasti, na ktorých sú farby trochu zmiešané a priesvitné. Tu bolo použité priemerovanie.



Obrázok 16. Výsledný obrázok susedia s parametrami 2/20/6.

Skúšaním bolo zistené, že najlepšie výsledky sa dosahujú pri použití parametrov 2/20/5, 2/20/6 a 2/20/7. Prvé dve čísla predstavujú nastavenie pre časť algoritmu prevzatého z rozmedzia. Vidieť, že sa zhoduje s optimálnymi parametrami tejto funkcie. Posledné číslo je minimálny počet zhôd v okolí aktuálneho bodu. Pri iných nastaveniach sa fotografia ešte zhorší vo viacerých smeroch. Výrazné farebné odtiene blízkych oblastí, viac duchov a slabé odstránenie nežiaducich objektov.



**Obrázok 17. Funkcia susedia s parametrami 4/20/7 už neodstráni niektoré objekty. Ale pri tomto nastavení je úspešne zobrazená obloha.**

Tiež treba spomenúť časovú náročnosť algoritmu. Výsledná fotografia sa počíta veľmi dlho. Použitie rovnakej sady vzorových fotografií ako pri predchádzajúcich funkciách spôsobuje počítanie až v rádoch desiatok minút. Aj preto je táto funkcia vo výslednom programe použitá iba pri spustení s daným parametrom.

## 7 Záver

Cieľom práce bolo zrekonštruovať pozadie z viacerých fotografií. Pozadím sa rozumie to, čo je nehybné, a malo by sa nachádzať na fotografovanej scéne vždy. Ak sa časť pozadia na fotografii nenachádza, znamená to, že nejaký objekt ju prekryl. Úlohou práce bolo detekovať tieto objekty a následne ich odstrániť. Výslednému programu sa to darí dosiahnuť. Existujú prípady, ktorých výsledok nedosiahol úplne dokonalú fotografiu. Stopercentne úspešnú rekonštrukciu pozadia prakticky ani nie je možné dosiahnuť. Práca, ktorá by sa jej čo najviac priblížila, vyžaduje nepomerne väčšie časové rozpätie na podrobnejšie bádanie a objavovanie nových postupov.

Pri skúšaní úspešnosti rôznych algoritmov sa vyskytli aj niektoré problémy. Napríklad na správne zobrazenie oblohy bolo treba pokusne získať vhodné parametre algoritmov. Tiež sa stáva, že výsledná fotografia obsahuje ľudí, ktorí sa za pozadie nepovažujú. Sú to osoby, ktoré sa nachádzajú na každej fotografii na tom istom mieste. Vtedy ich program pokladá za pozadie, pretože sú nepohybliví.

Výsledný program zobrazí správne aj fotografie, ktoré boli získané za rôzneho osvetlenia. Dosiahne sa to použitím správnej kombinácie použitých metód.

Na základe dosiahnutých výsledkov možno skonštatovať, že práca splňuje požiadavky zadania.

Prípadné doplnenie programu v budúcnosti by mohlo byť detekovanie pohybu scény. Ak by nebolo nutnosťou používať statív a mohlo sa fotografovať aj z ruky, určite by to zlepšilo použiteľnosť programu. Túto vlastnosť práca neobsahuje, pretože sa jedná o náročné zohľadnenie optickej sústavy a následnej korekcie chýb. Samotná implementácia v dostatočnej kvalite by zabrala samostatnú prácu.

# Literatúra

- [1] KaewTraKulPong, P., Bowden, R.: An improved adaptive background mixture model for real-time tracking with. Proc. European Workshop Advanced, 2001: s. 1-5.
- [2] Grimson, W., Stauffer, C., Romano, R.: Using adaptive tracking to classify and monitor activities in a site. IEEE Comput. Soc. 1998, 1998.
- [3] Friedman N., Russell, S.: Image segmentation in video sequences: A probabilistic approach. The Thirteenth Conference on Uncertainty in Artificial Intelligence, 1997.
- [4] Criminisi, A., Perez, P., Toyama, K.: Object removal by exemplar-based inpainting. IEEE Computer Society Conference, 2003: s. 1-8.
- [5] Efros, A., Leung, T.: Texture synthesis by non-parametric sampling. Proc. ICCV, 1999: s. 1033-1038.
- [6] Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning lowlevel Vision. Int. J. Computer Vision, 2000: s. 25-47.

# **Zoznam príloh**

Príloha 1. CD